



US006144942A

United States Patent [19]
Ruckdashel

[11] **Patent Number:** **6,144,942**
[45] **Date of Patent:** **Nov. 7, 2000**

[54] **METHOD FOR NOTIFYING AN INDIVIDUAL OF A PREVIOUSLY SCHEDULED EVENT**

[75] Inventor: **Glenn A. Ruckdashel, Boise, Id.**

[73] Assignee: **Micron Electronics, Inc., Nampa, Id.**

[21] Appl. No.: **09/067,344**

[22] Filed: **Apr. 28, 1998**

[51] Int. Cl.⁷ **G06F 17/60**

[52] U.S. Cl. **705/9; 705/8; 705/7; 705/1**

[58] Field of Search **705/8, 9, 7, 1**

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,530,861	6/1996	Diamant et al.	705/8
5,606,695	2/1997	Dworzecki	705/8
5,623,413	4/1997	Matheson et al.	701/117
5,657,453	8/1997	Taoka et al.	705/1
5,706,200	1/1998	Kumar et al.	700/100
5,721,913	2/1998	Ackroff et al.	707/103
5,737,728	4/1998	Sisley et al.	705/8
5,742,931	4/1998	Spiegelhoff et al.	705/8
5,799,286	8/1998	Morgan et al.	705/30

5,826,040	10/1998	Du et al.	705/8
5,855,006	12/1998	Huemoeller et al.	705/9
5,860,067	1/1999	Onda et al.	705/9
5,867,824	2/1999	Saito et al.	705/9
5,872,926	2/1999	Levac et al.	340/825.44
5,890,131	3/1999	Ebert et al.	705/7
5,893,074	4/1999	Hughes et al.	705/8
5,902,352	5/1999	Chou et al.	709/102
5,907,490	5/1999	Oliver	700/900
5,913,201	6/1999	Kocur	705/9
5,918,207	6/1999	McGovern et al.	705/1
5,943,652	8/1999	Sisley et al.	705/9
5,948,054	9/1999	Nielsen	709/200

Primary Examiner—Allen R. MNacdonald

Assistant Examiner—Akiba Rabina Boyce

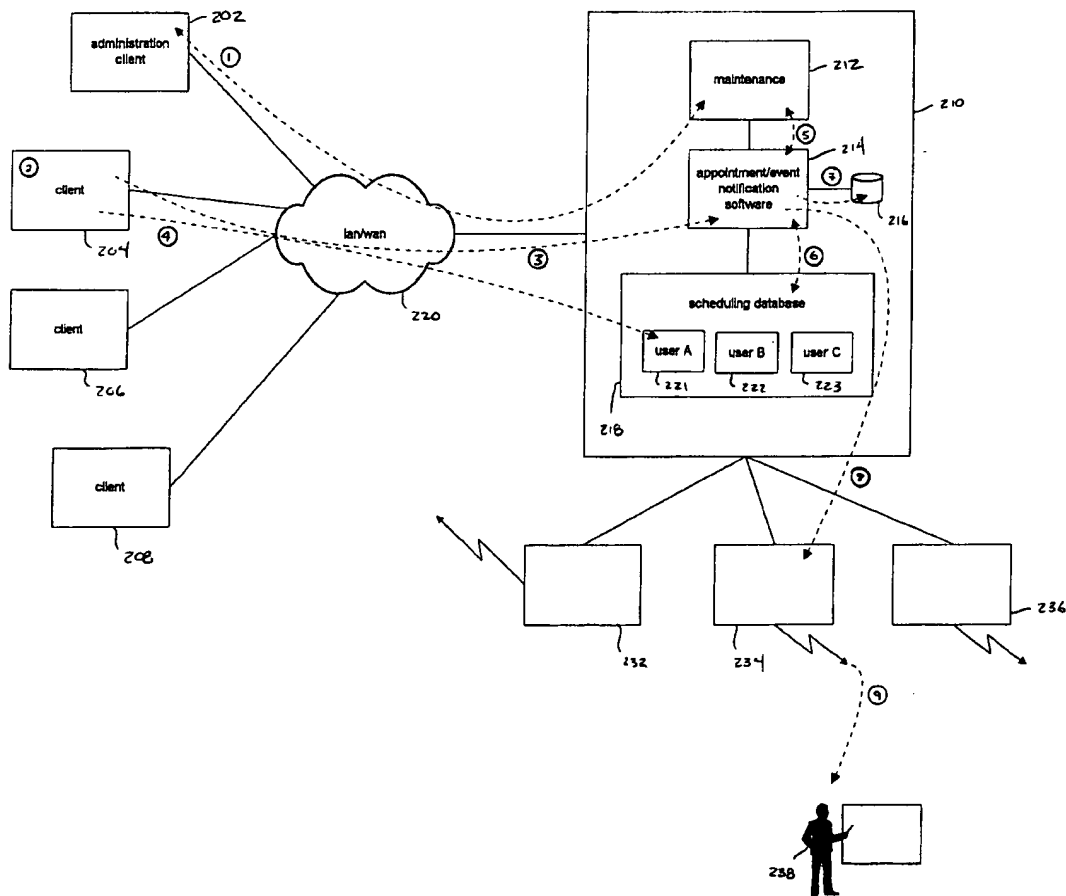
Attorney, Agent, or Firm—Dorsey & Whitney LLP

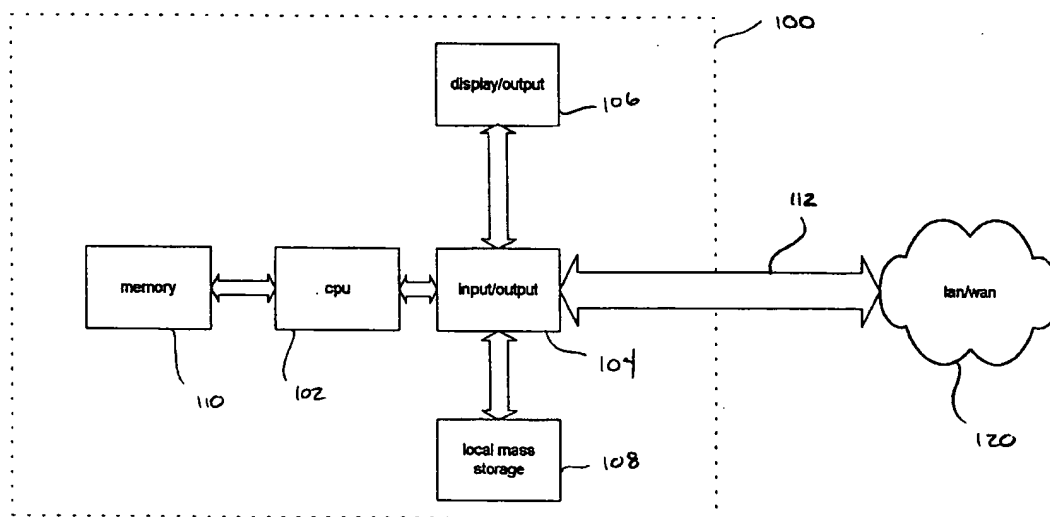
[57]

ABSTRACT

A method for notifying an individual of a previously scheduled event. The individual's scheduling information is accessed, and analyzed. When a scheduled event arrives, the user is notified in the manner that the user specifies. Methods of notification include wireless pager, email, and wireless telephone.

27 Claims, 11 Drawing Sheets



**Figure 1**

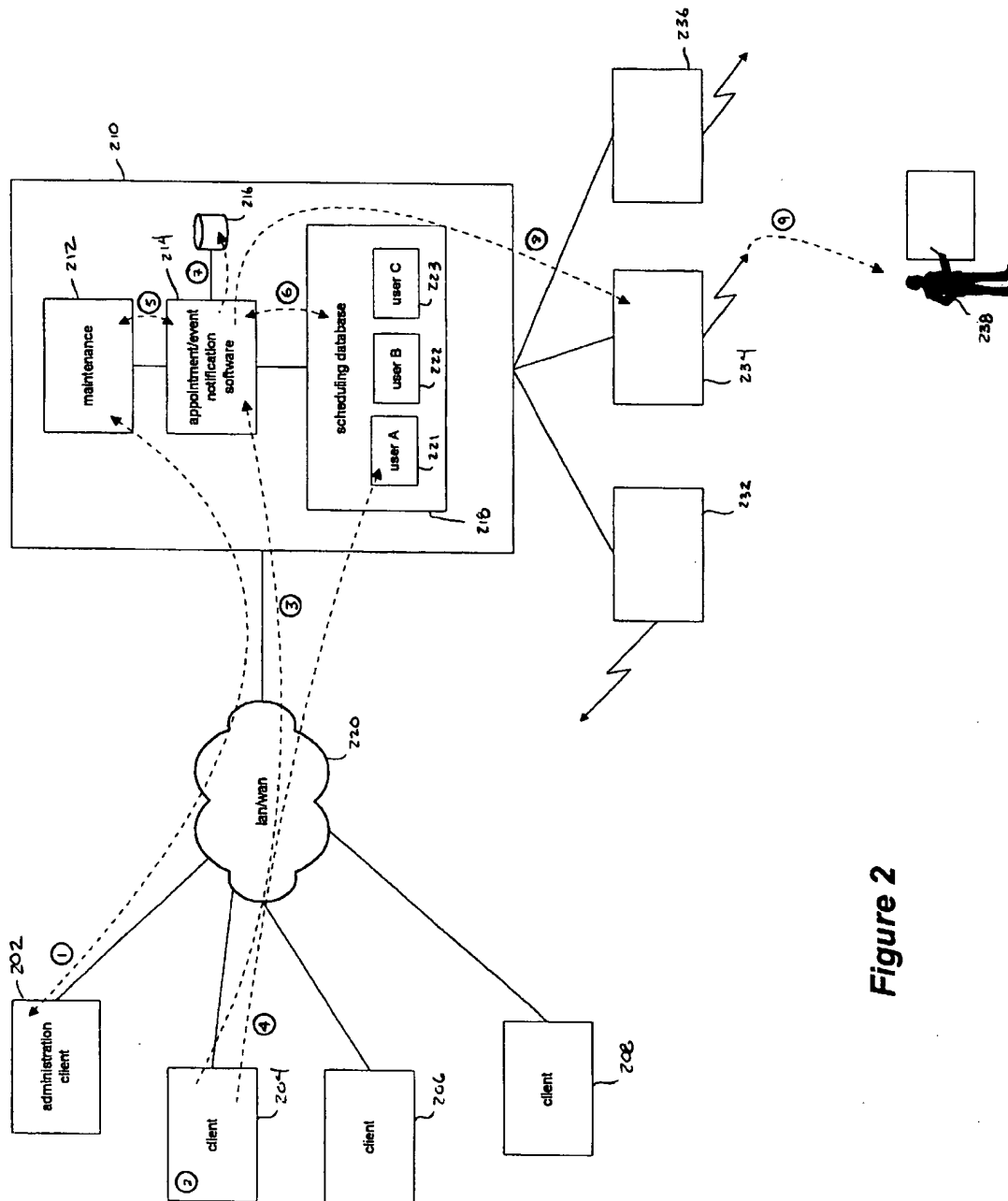
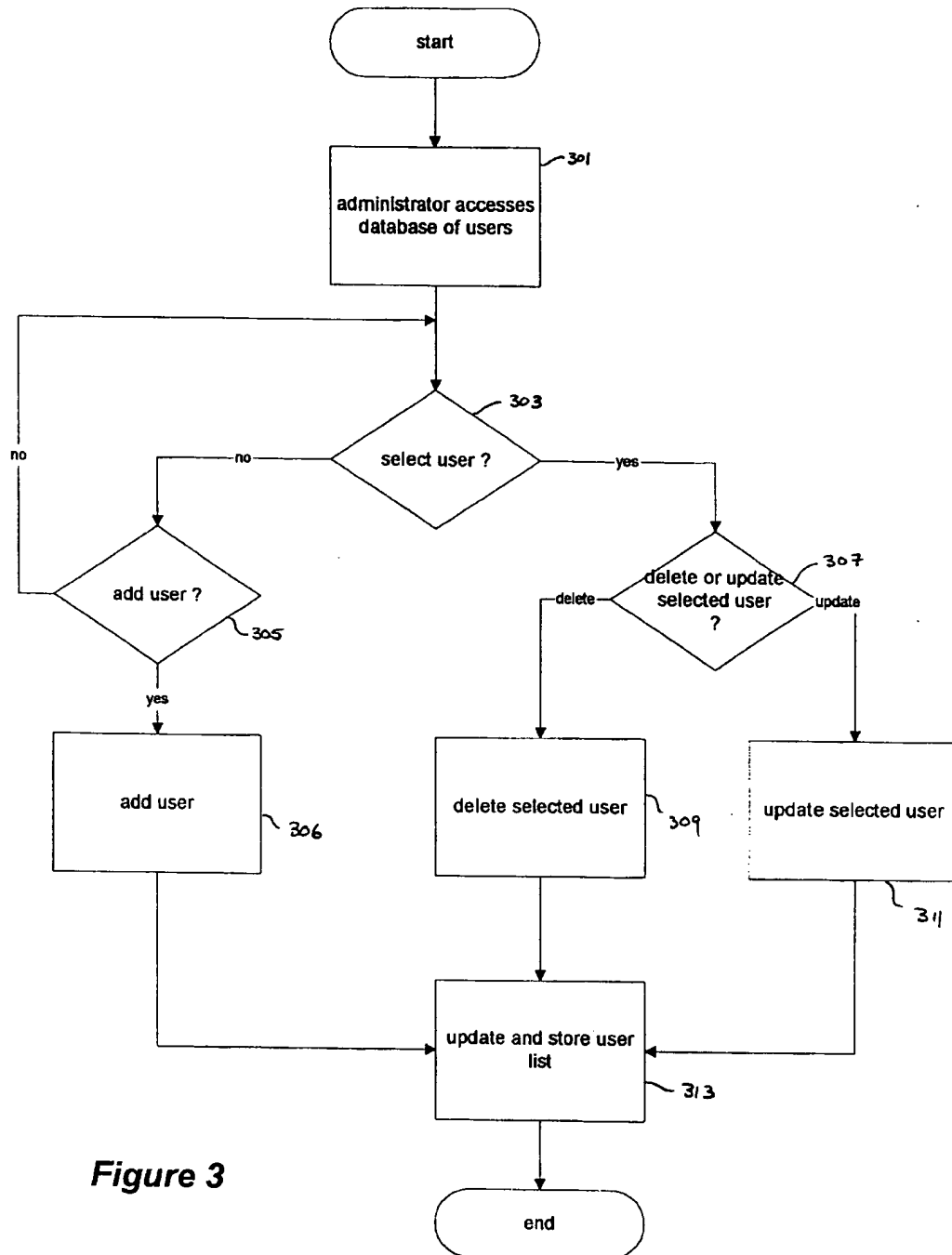
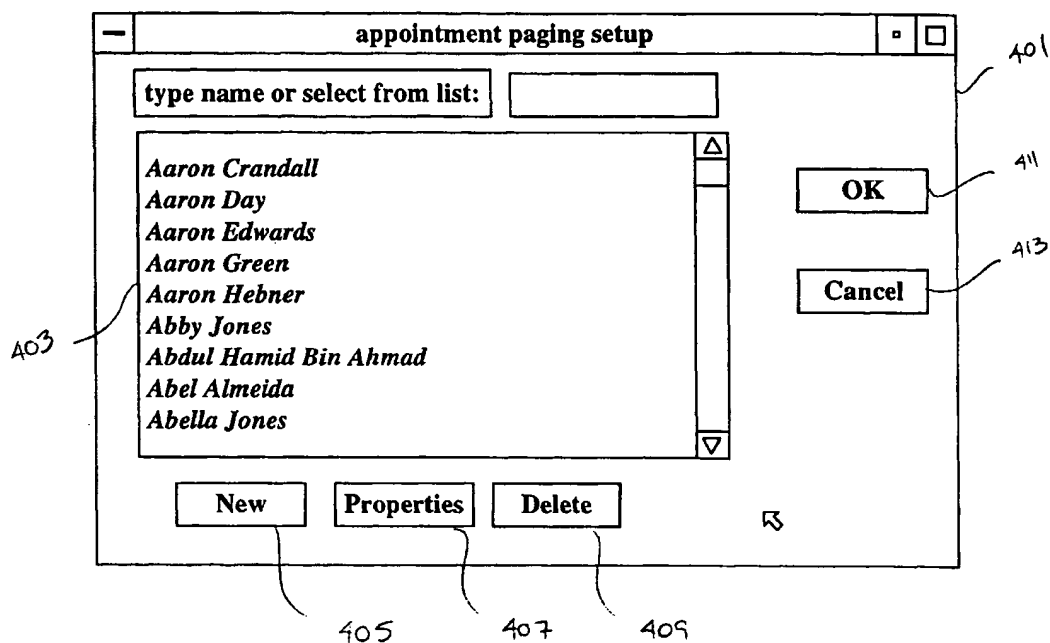
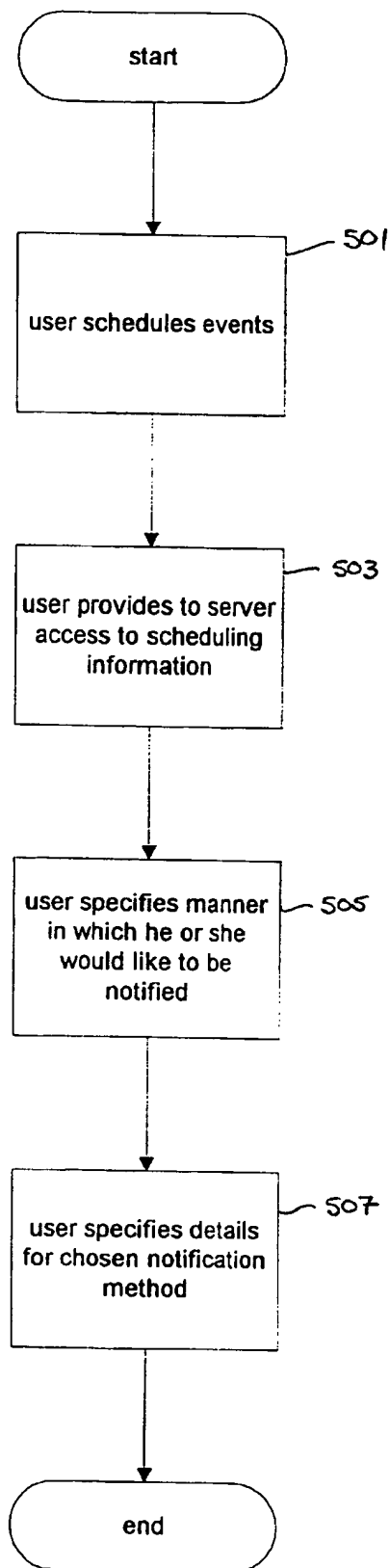


Figure 2

**Figure 3**

**Figure 4**

**Figure 5**

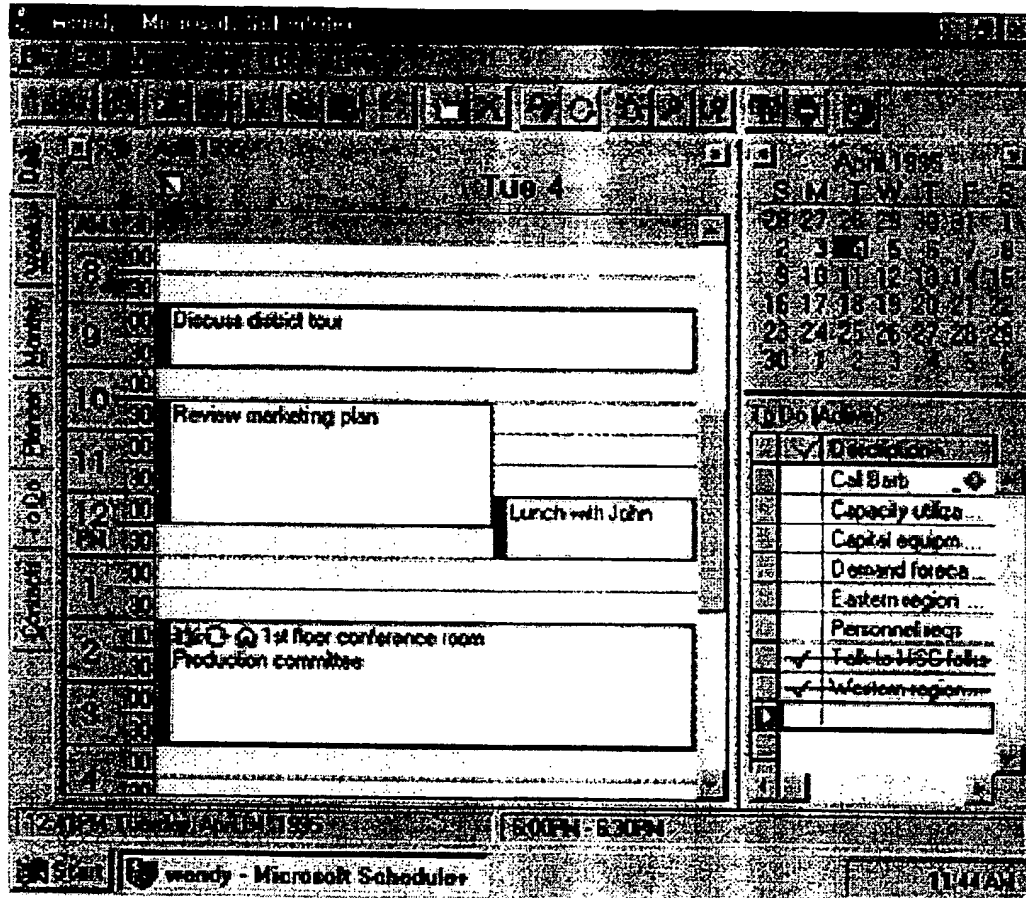


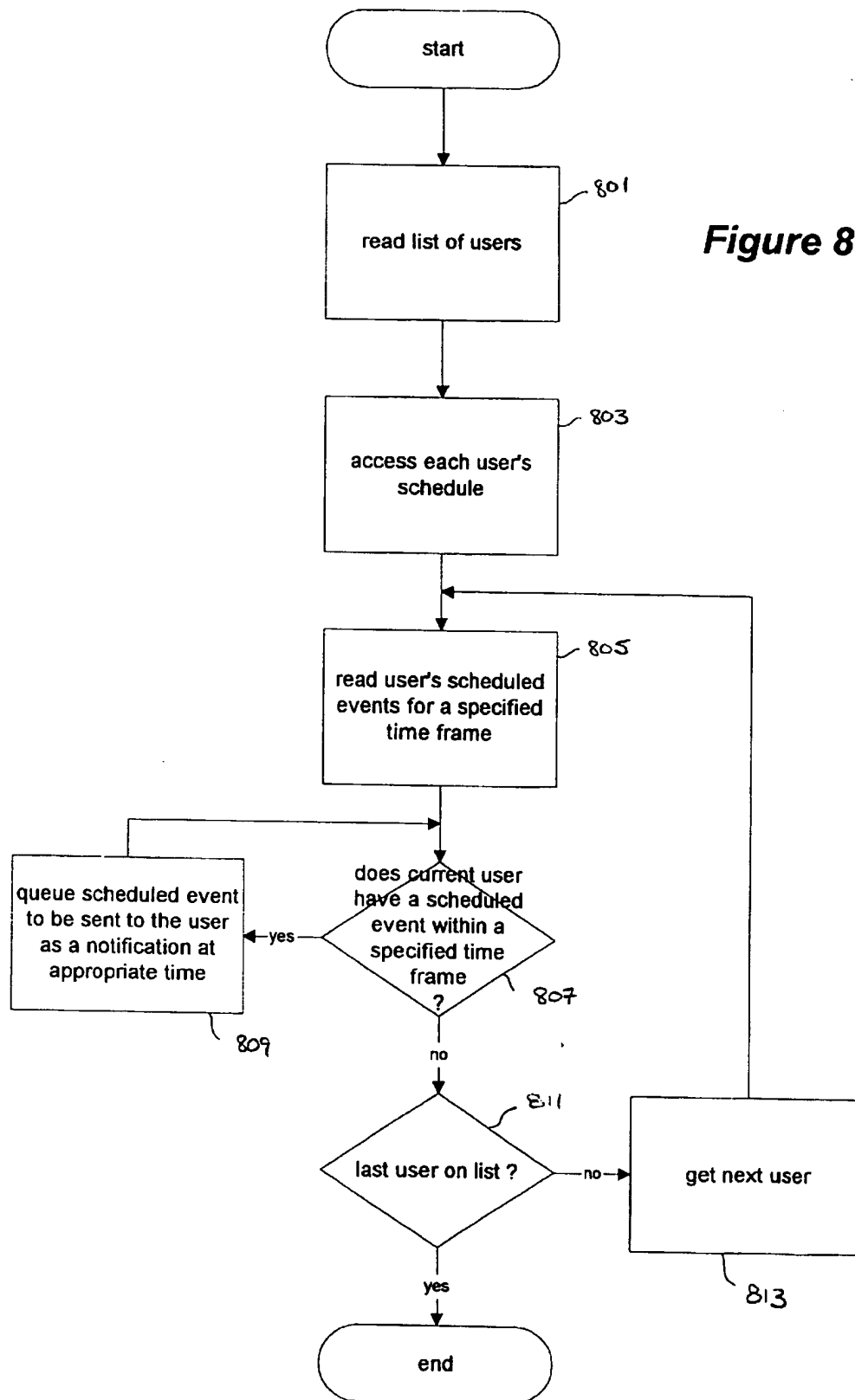
Figure 6

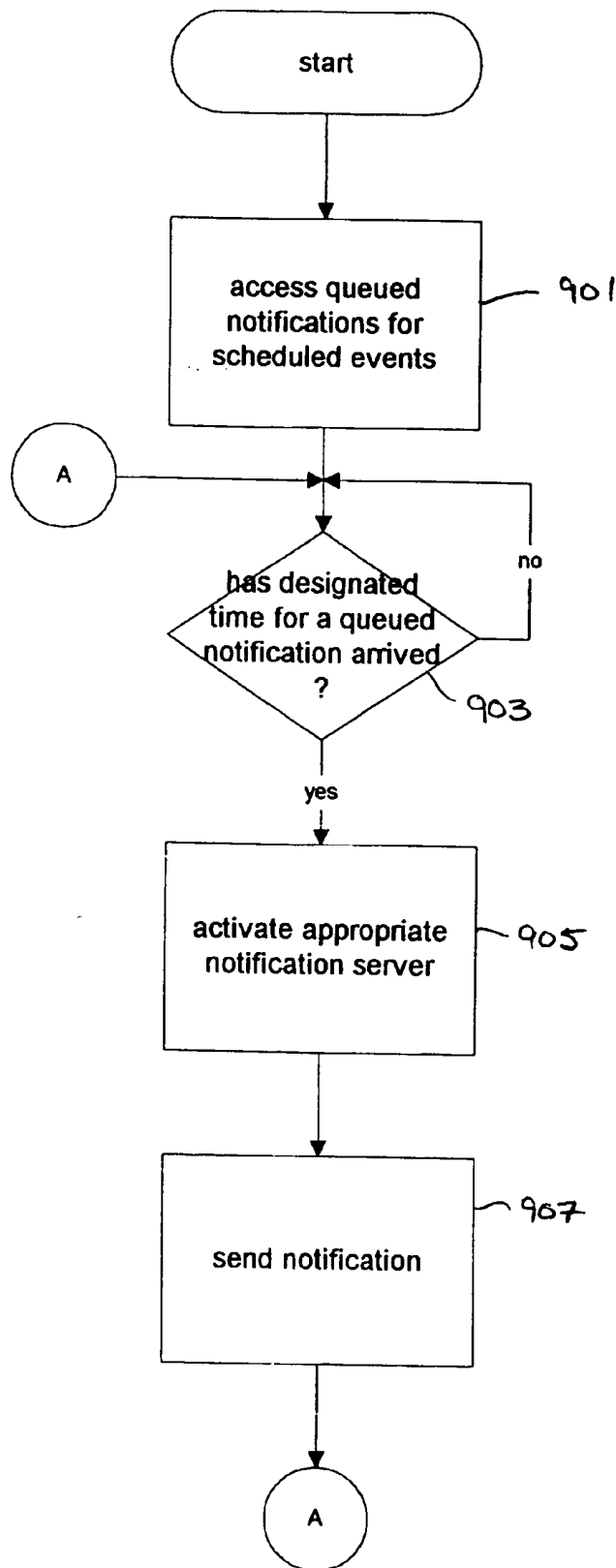
A screenshot of a "notification configuration" dialog box. The title bar contains a minus sign, the text "notification configuration", and standard window control buttons. The main content area displays the event title "Appointment: Production Committee Meeting" (labeled 717) and the question "how would you like to be notified of this event?". Below this, three notification options are listed, grouped by a bracket (labeled 703):

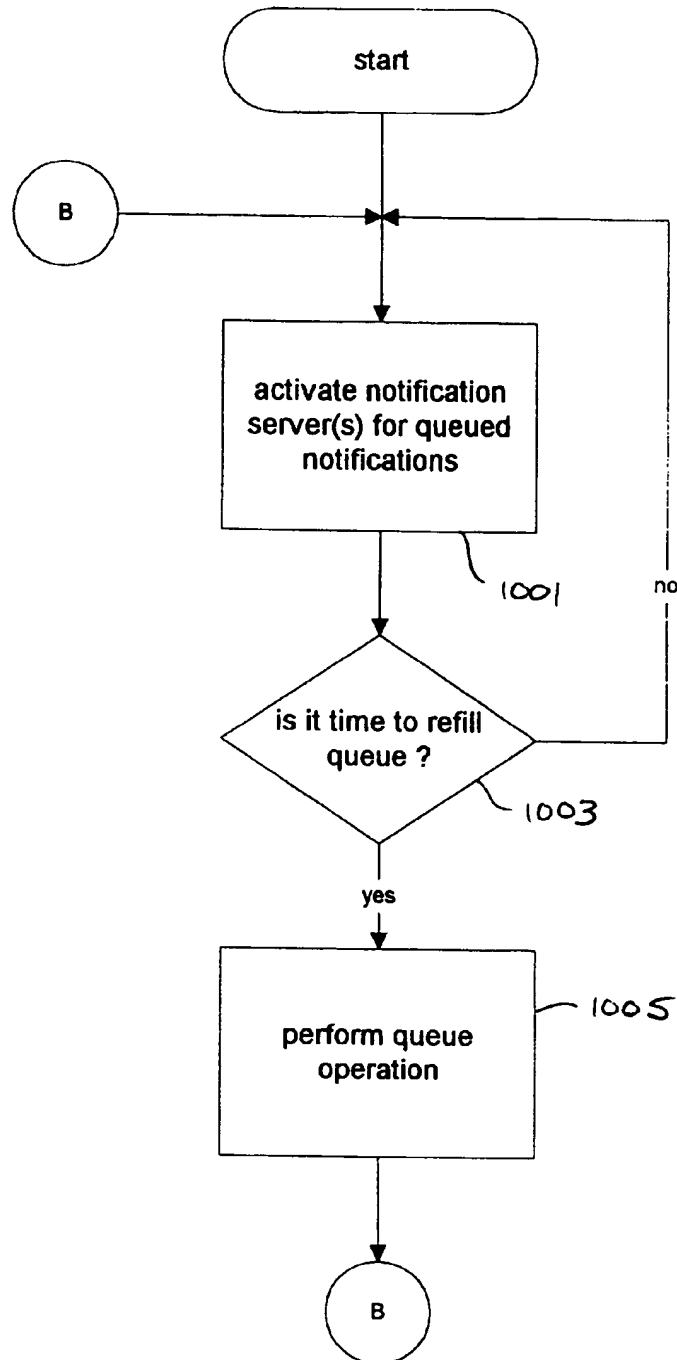
- "email; address:" (labeled 713) with an empty text field (labeled 709) and an unchecked checkbox.
- "wireless messaging device/pager:" (labeled 715) with an empty text field (labeled 711) and an unchecked checkbox.
- "cellular phone; number:" (labeled 705) with a text field containing "790-3205" (labeled 707) and a checked checkbox.

At the bottom of the dialog are three buttons: "OK", "Other", and "Cancel". A small icon is located in the bottom right corner of the dialog area. The entire dialog box is labeled 701.

Figure 7

Figure 8

**Figure 9**

**Figure 10**

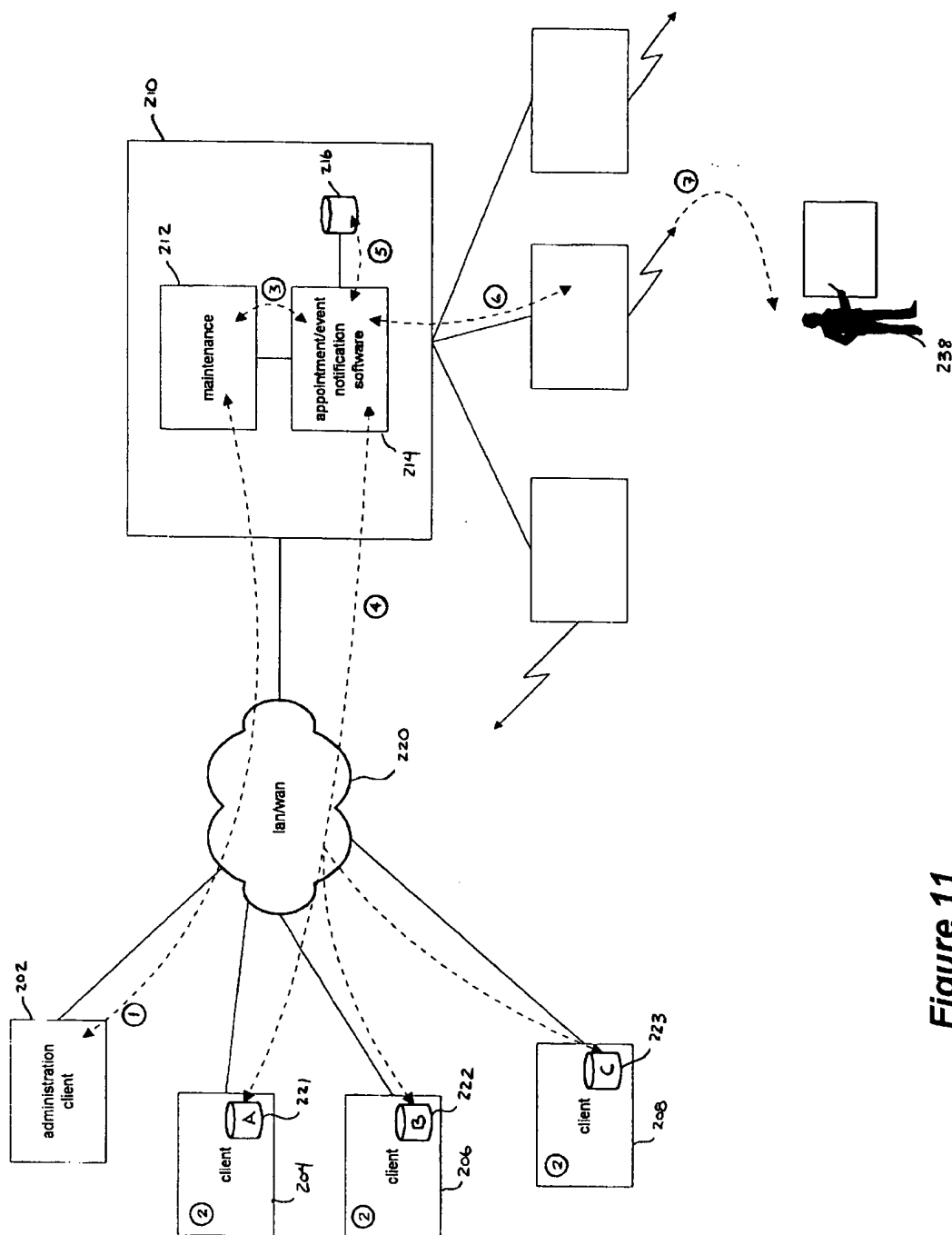


Figure 11

METHOD FOR NOTIFYING AN INDIVIDUAL OF A PREVIOUSLY SCHEDULED EVENT

FIELD OF THE INVENTION

The present invention relates generally to techniques for notifying individuals of scheduling or meeting information. More particularly, the present invention relates to techniques for integrating the use of communication devices and methods with scheduling information.

BACKGROUND OF THE INVENTION

The proliferation of personal computers and portable information devices has provided substantial improvements in the ability to track appointments and communicate with others. For example, many individuals now store information on a personal computer in a way that allows them to track projects, work assignments, and deadlines. And as portable personal computers have become more powerful and inexpensive, scheduling software has become popular and widely used.

At the same time, wireless paging devices, cellular phones, and other portable communication devices are also becoming widely used. These devices allow others to contact or communicate with a person who might otherwise be unavailable. Because communication has become so much easier, people who use modern communications devices are much more free to leave their traditional office environment or home because they know that they can be reached by those who wish to reach them.

While use of the use of scheduling software has become very common, and while the use of new types of communication devices has also become very common, there has been no efficient interaction between the two. Previously, there has been no effective integration of scheduling and communications capabilities, and no way to take advantage of the benefits available if such integration were possible. For example, one possible benefit is that the scheduling system itself can use a communication device to ensure that an individual is aware of important events on his or her schedule. Thus, there exists a need for an efficient system and method for efficiently integrating scheduling software with a variety of communication devices.

SUMMARY OF THE INVENTION

The present invention relates to techniques for issuing device requests by proxy. In one embodiment of the invention, the invention relates to a method for communicating scheduling information to a user. Specifically, such a method comprises the acts of: performing scheduling tasks at a client computer; storing the user's scheduling information; accessing the user's scheduling information; analyzing the user's scheduling information; queuing to a storage area events in the scheduling information that are within a specified time frame; accessing the storage area to retrieve the queued events; and initiating communication with the user at or about the time that the queued event is to occur to inform the user of a scheduled event when the user is not at the client computer.

In another embodiment of the present invention, the present invention relates to a method for communicating scheduling information to a user that operates scheduling software at a client computer on a network. The scheduling information for the user is stored at the client computer. The method in such an embodiment comprises the acts of: initiating a request over the network to retrieve scheduling

information for the user; accessing the user's scheduling information through API calls to the scheduling software on the client computer; retrieving at least some of the user's scheduling information; analyzing the user's scheduling information; queuing to a storage area information relating to events in the user's scheduling information; accessing the storage area to retrieve the queued events; and initiating communication with the user at or about the time that the queued event is to occur so as to inform the user of a scheduled event when the user is not at the client computer.

Additional embodiments and features, and the nature of the present invention may be more clearly understood by reference to the following detailed description of the invention, the appended claims, and to the several drawings herein.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a computer system having central processing unit that is connected to a local or wide area network through an input-output module.

FIG. 2 is a block diagram of a system for notifying an individual of a previously-scheduled event, where the scheduling information is stored at the server.

FIG. 3 is a flow chart of the operation of a program used by an administrator of the system.

FIG. 4 is a representation of a window associated with an administration program shown on a screen display.

FIG. 5 is a flow chart illustrating the operation of a program that allows a scheduling entry for a given event to be configured for later notification.

FIG. 6 is a representation of a window on a screen display, where the window is associated with a scheduling program executing at the client computer.

FIG. 7 is a representation of another window associated with a scheduling program executing at the client computer, where the window allows the user to configure the manner in which he or she is notified for the scheduled event.

FIG. 8 is a flow chart illustrating the operation of an operation carried out by the notification software.

FIG. 9 is a flow chart illustrating the operation of another operation carried out by the notification software.

FIG. 10 is a flow chart illustrating the operation of the notification software.

FIG. 11 is a block diagram of a system for notifying an individual of a previously-scheduled event, where the scheduling information is stored at each individual's client computer.

DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 is a block diagram of a computer system 100 having a central processing unit 102, and an input/output module 104. The processor 102 is connected to the memory 110 and the input/output module 104. The module 104 includes circuitry for interfacing the processor 102 with other devices within the computer system 100, including the display or output device 106, and the (optional) local mass storage 108. The module 104 also interfaces the computer 100 to the network 120, which may be a local or wide area network. Typically, the network is a TCP/IP internetwork, but any other network protocol could be used.

FIG. 2 shows a block diagram of a system having a number of client computers 202, 204, 206, and 208, which may be of the type illustrated in FIG. 1. Each of these client

computers is on a network 220, which may be either a wide area or local area network. Also shown is a computer 210, which may act as a server for some functions associated with the client computers. The computer 210 includes a maintenance module 212, appointment notification software 214, and a queue database 216. Also included within the computer 210 is a scheduling database 218, having scheduling information for at least three different users: A, B, and C. The scheduling information for each of these three users is stored in modules 221, 222, and 223, respectively. It should be understood that one or more of the modules within the computer 210 could be implemented as separate computers. For example the maintenance module 212 and appointment notification software could be implemented on or executing on a separate computer.

Notification servers 232, 234, and 236 are also shown in FIG. 2. Each of these servers provides a way of communicating with individuals, such as the individual represented by numeral 238. These individuals may be in their office (or home), near their office, or far away from their office. The notification servers 232, 234, and 236 have the capability, however, of communicating with individuals wherever they might be. For example, notification server 232 is an email server that is used to communicate via email, which can be sent to the user's normal desktop computer, or to another computer at which the user may be working. This notification server 232 may be connected back into the network 220 so that the email messages may be delivered through the network.

Notification server 234 is a wireless telephone server that is capable of initiating cellular or other wireless audio communication to a given telephone number. Normally, the user 238 carries with him or her a portable telephone. Notification server 236 is a wireless messaging server or wireless paging server that is used to send, through a wireless mechanism, messages and/or other information. As is well known in the art, individuals wear a paging receiver, and are notified when a message has been sent.

In some embodiments, the notification servers are not integrated with the computer 210, and the notification software 214 is not integrated with any of the software controlling the notification servers. In these situations, computer 210 communicates with the notification servers 232, 234, and 236 through an interprocess communication mechanism. Such a mechanism may be a custom application programming interface made available by the notification servers 232, 234, and 236 to other programs, such as the notification software 214. Such a mechanism could also involve the use of a component technology, such as Microsoft's ActiveX and/or OLE (formerly Object Linking and Embedding) technologies based on the Component Object Model and Distributed COM (formerly Network OLE) architectures, Object Management Group's CORBA technologies (e.g., Internet Inter-Object Protocol (IIOP)), JavaBeans component technology, or Remote Method Invocation technology available in a Java environment. It should be understood, however, that any other method or system for interprocess communication now known or hereinafter developed may be appropriate for use in practicing the invention.

In the system of FIG. 2, a system administrator will typically perform system administration tasks, using administration software executing at the administration client 202. Users A, B, and C perform scheduling tasks using scheduling software executing at the client computers, 204, 206, and 208, respectively. In the embodiment of FIG. 2, the database of appointments, deadlines, and other information for users

A, B, and C is stored at the server computer 210 within the scheduling database 218. The client computers 204, 206, and 208 used by these users may be desktop machines located in their offices, or these computers may be these users' regularly-used mobile computers. Users A, B, and C may often be away from their computer, however, so they may be unable to use the computer to check on scheduling information and watch for upcoming events. The system of FIG. 2 is designed to permit effective use of previously scheduled information in such a situation.

The appointment notification software 214 executes on the computer 210. This software retrieves information from the scheduling database 218, and then interacts with the notification servers 232, 234, and 236 to notify one of the users A, B, and C of upcoming events on their schedules. The notification software 214 may access data in the scheduling database through a customized query protocol. The data may also be accessible through a known query protocol, such as through the Structured Query Language and/or the Open Database Connectivity (ODBC) standard. The data may also be stored in a format that permits retrieval through an interprocess communication mechanism, such as an ActiveX/OLE interface, DDE, through a custom API which may be implemented through a dynamic link library, or through any of the other mechanisms previously mentioned. Similarly, other communication that the notification software carries out with other modules or devices, such as the maintenance module 212 or one of the notification servers 232, 234, and 236, are through an interprocess communication mechanism, a directory protocol, through a data retrieval protocol, or through some other mechanism.

FIG. 3 is a flow chart illustrating the operation of an administration program that executes at client computer 202 in FIG. 2. The administration program is used by a system administrator to add, delete, and modify the list of people that use the appointment notification software. The program may also modify the attributes of a particular individual's entry, such as that individual's cellular telephone number, email address, wireless pager number, or other information. As shown at 301 in FIG. 3, the administration software accesses the database of users, which may be stored locally, or at another location on the network, as shown at 212 in FIG. 2. At 303, if the administrator has selected a user, the administrator can update at 311 or delete at 309 the selected user. Otherwise, the administrator is given the option to add a user to the system at 305. Once the desired task has been performed, the maintenance database is updated and stored. In FIG. 2, the database 212 is maintained at computer 210.

FIG. 4 is a representation of a window 401 taken from a screen display that might be seen by the administrator when operating the administration program executing on the client computer 202. The current list of users is shown in box 403. The administrator scrolls through the list of users, and selects a user. The administrator then selects either the properties button 407 or the delete button 409. Selection of the properties button 407 permits the user to modify the attributes of the selected user. Selection of the delete button 409 removes the selected user from the list, and thus also removes the user's ability to use the system to notify him or her of scheduled events. A new user can be added to the list by selecting the new button 405. When the administrative tasks are complete, the administrator closes the window 401 by selecting either the OK button 411 to save the changes, or the Cancel 413 button to discard any changes.

FIG. 5 is a flow chart illustrating the operation of the scheduling software, such as that used by users A, B, and C in FIG. 2. Most often, this software executes on the client

5

computer, but in the embodiment of FIG. 2, the scheduling information is stored on the server 210. At 501 in FIG. 5, the user interacts with the scheduling software, and schedules appointments, deadlines, and other events of interest. FIG. 6 is a representation of a screen display of such a program, showing scheduling or appointment data for Tuesday, April 4. The scheduling software shown in the Figure is Microsoft Schedule+, which is a scheduling program that may be used with the present invention.

At 503 in FIG. 5, the user provides access to his or her scheduling information to the appointment notification software executing on the server 210. This is necessary to allow the appointment notification software to read at least some of the entries in that person's schedule, and thus notify that person as the time for the event approaches. Access can be provided by sending a command to the notification software, by configuring the scheduling software so that other programs can access it, or by using a password scheme. In another embodiment, the administrator may regulate the access to each user's scheduling information.

At 505 and 507 in FIG. 5, the user configures how he or she wishes to be notified for a particular appointment. For example, FIG. 7 is a representation of a window 701 that may be associated with the scheduling program shown in FIG. 6. As shown in FIG. 7, by selecting one (or more) of the boxes 703, the user can select the manner in which he or she is notified of the event or appointment 717 identified at the top of the window 701. In FIG. 7, the user has selected box 705, which specifies that the user is to be notified of the "production committee meeting" by cellular telephone. The user also specifies the desired telephone number in box 707. Boxes 713 and 715 relate to other methods (email and wireless messaging device or pager) of notifying the user as the specified appointment approaches. Boxes 709 and 711 also allow the user to specify an email address or pager number/identifier associated with those methods.

Although the configuration options shown in FIG. 7 are for a particular appointment, it should be understood that similar configuration settings may be available on a global basis, so that all appointments in a given schedule have a default notification configuration. A window such as that shown in FIG. 7 may be used to alter the default configuration for a particular appointment or notification. It should also be understood that other methods for configuring a particular notification or groups of notifications may be employed. Further, it should be understood that other methods for notifying users of events, beyond those shown in FIG. 7, could be employed.

FIG. 8 is a flow chart of an operation carried out by the notification software 214 executing on server 210 in FIG. 2. At 801 in FIG. 8, the notification software represented by 214 in FIG. 2 accesses the list of users from the maintenance module 212. This access will typically involve simply reading the file or directory of users maintained by the administrator. In some embodiments, this access may involve use of directory protocols, e.g., LDAP or NDS, or the use of a database query language. Thereafter, at 803 and 805 in FIG. 8, the notification software 214 accesses each user's scheduling information stored in the database 218, and reads the scheduled events in each user's schedule for a specified time frame. The specified time frame will typically be a period of time starting at the present and extending into the future. Such a period may be, for example, a period of eight hours starting from the present time. This allows the notification software 214 to analyze the scheduling information for all approaching events, and prepare to notify the user of these events. In some embodiments, scheduling

6

information that is well into the future may not be retrieved or read from the database 218 because users often are not interested in being notified of events until the scheduled time is near. It should be understood, however, that in other embodiments, various applications of the invention and other considerations may make a different time frame (or time frames) more appropriate.

If at 807 an analysis of the scheduling information indicates that the user has an event scheduled for the specified time frame, and if the user wishes to be notified of the event, the scheduled event or events are queued at 809 to be sent to the user as a notification at the appropriate time. This process repeats until the scheduling information for the last user on the list has been analyzed at 811.

FIG. 9 is a flow chart of another operation carried out by the notification software 214 executing on the server 210 in FIG. 2. At 901, the notification software 214 accesses the queued notifications for scheduled events that were queued at 809 in FIG. 8. Once the queued operations are accessed, the notification software waits until a designated time for a queued notification arrives. This designated time may be at the time of the scheduled event, but it is preferred that the designated time be some small period of time before the event so that the user has a better opportunity to prepare for the event or get to the event at the scheduled time. The amount of this lead time can be configured globally, or configured by each individual user, and perhaps for each individual event.

Once a designated time for a queued notification arrives, the notification software 214 activates the appropriate notification server at 905. For example, if the notification entry for the event is configured so that the user is notified by wireless telephone, the notification software sends a command to the notification server 234, or otherwise causes that notification server 234 to place a telephone call to the user at 907.

Thus, as described in FIG. 8, the notification software periodically reads scheduling information from the scheduling database, and queues events scheduled for the near future to be sent to a notification server at an appropriate time. And as described in FIG. 9, once the designated time for one of these queued events arrives, the appropriate notification server is activated, and the notification is sent. FIG. 10 illustrates these processes in one embodiment of the present invention. At 1001, and at the appropriate times, the notification software activates the appropriate notification server for the (previously) queued notifications. At 1003, the notification software determines whether the appropriate time to refill the queue has arrived. If so, the queue is refilled at 1005, and the notification software goes back to activating the notification server(s).

Referring again to FIG. 2, the operation of the previously-described embodiment of the invention is illustrated. As described in FIG. 3, the administrator performs administration tasks at the client computer 202, accessing the maintenance database or module 212. Once the desired user profiles are updated or otherwise modified, the new user list and associated information is stored in the maintenance database or module 212. This operation is represented in FIG. 2 by the broken arrow labeled "1" in FIG. 2.

User A regularly works with client computer 204 in carrying out day to day scheduling tasks. See "2" in FIG. 2. The other users, B and C, regularly work with computers 206 and 208. Each user also provides read access to the notification software, so that the notification software can read that user's schedule from scheduling database. In the

embodiment shown, this access is provided by configuring the maintenance database. Although this is a task that could be performed by the administrator, it is shown in FIG. 2 (for user A) as being carried out by the user at the client computer. See "3" in FIG. 2. And in connection with performing scheduling tasks, each user specifies details about the chosen notification method for that user's appointments, and this information is stored in the scheduling database 218. See "4" in FIG. 2.

The notification software periodically reads the list of users from the maintenance module 212 and accesses each user's scheduling information from the scheduling database 218. See "5" and "6" in FIG. 2. As scheduled events for a given user (e.g., user A) approach, they are queued by the notification software 214 for later delivery to one of the notification servers. See "7" in FIG. 2. And once the designated time for a specific event arrives, the notification software 214 activates the desired notification server, and user A (designated by 238), who may be in the field, traveling, or otherwise away from his or her normal office, is notified of the upcoming event. See "8" and "9" in FIG. 2. In the embodiment shown, the notification server 234 is used to notify user A. This notification server is hardware that dials user A's wireless telephone number, and plays an audio message. Such a message may be an audio rendition (i.e., reading) of the information associated with the user's scheduled event, or it could be any other useful message that notifies user A that the time for a previously scheduled event has arrived.

FIG. 11 is a diagram of a system in an alternate embodiment of the present invention. In FIG. 11, the scheduling information is stored at the client computer for each user, rather than at the server as in FIG. 2. For example, since computer 204 is typically used by user A, the scheduling information for user A is stored on storage device 221 within computer 204 in FIG. 11. Similarly, the scheduling information for users B and C are stored at computers 206 and 208, respectively.

The operation of this alternate embodiment of the present invention is illustrated in FIG. 11. As was the case with FIG. 2, the administrator performs administration tasks at the client computer 202, accessing the maintenance database or module 212, and storing the new user list and associated information in the maintenance module 212 at the server 210. See "1" in FIG. 11.

Users A, B, and C carry out scheduling operations at the client computers 204, 206, and 208, while also providing the specifics about how to be notified for any scheduled appointments or events. In addition, each user also provides read access to the notification software, so that the notification software can read that user's schedule from scheduling database. In the embodiment of FIG. 11, this is done by configuring the scheduling program at the client computer to provide read access to the scheduling data stored at the client. See "2" in FIG. 11.

The notification software periodically reads the list of users at the server 210 from the maintenance module 212. See "3" in FIG. 11. The notification software then accesses, over the network, each user's scheduling information. See "4" in FIG. 11. This access can be carried out through a customized interface or API, perhaps available remotely through a Dynamic Link Library, COM, ActiveX/OLE, and/or DCOM interface on the Windows platform. In other embodiments, this access could be through a protocol such as IIOP, through Java-related technologies (e.g., JavaBeans or Java RMI calls), or even a through high-level Internet

protocol such as HTTP. However, it should be understood any method or system for network communication now known or hereinafter developed may be appropriate for use in practicing the invention.

As scheduled events for a given user approach, they are queued by the notification software 214 for later delivery to the notification server. See "5" in FIG. 11. And once the designated time for a specific event arrives, the notification software 214 activates the desired notification server. See "6" in FIG. 11. This activation causes the user A (238) to be notified of the upcoming event. See "7" in FIG. 11.

As will be appreciated by one skilled in the art, the manner in which the maintenance module, scheduling information or scheduling database is stored or distributed on a network can be changed without departing from the spirit of the invention. Further, although the present invention has been described in terms of certain embodiments, various changes and modifications, even if not shown or specifically described herein, are deemed to lie within the spirit and scope of the invention and the appended claims. Accordingly, it is the appended claims, and not the foregoing illustrations and descriptions that define the scope of the invention.

What is claimed is:

1. A method for communicating scheduling information about a future event to a user, the method comprising the acts of:

performing scheduling tasks at a client computer, including configuring the manner in which the user is to be notified of the future event and the time at which the user is to be notified of the future event;

storing the user's scheduling information; accessing the user's scheduling information;

analyzing the user's scheduling information; queuing to a storage area scheduling information about the future event if the future event is within a specified time frame;

accessing the storage area to retrieve the scheduling information; and

initiating communication with the user to inform the user of the future event in the manner configured and at the time configured when the user is not at the client computer.

2. The method of claim 1, wherein the act of storing the user's scheduling information includes the act of storing scheduling information at the client computer.

3. The method of claim 1, wherein the act of storing the user's scheduling information includes the act of storing scheduling information at a server computer.

4. A method for communicating scheduling information to a user, the method comprising the acts of:

performing scheduling tasks at a client computer;

storing the user's scheduling information;

accessing the user's scheduling information;

analyzing the user's scheduling information;

queuing to a storage area events in the scheduling information that are within a specified time frame;

accessing the storage area to retrieve the queued events; and

initiating communication with the user at or about the time that the queued event is to occur to inform the user of a scheduled event when the user is not at the client computer,

wherein the act of accessing the user's scheduling information includes the act of accessing information about how the user is to be notified when an event approaches,

9

wherein the information about how the user is to be notified when an event approaches applies to a subset of all events in the user's scheduling information.

5. The method of claim 4, wherein the act of queuing to a storage area events in the scheduling information that are within a specified time frame includes the act of queuing to a storage area events that are scheduled for a time within twenty-four hours of the act of accessing the user's scheduling information.

6. The method of claim 5, wherein the act of initiating communication with the user includes the act of placing a wireless telephone call to the user.

7. The method of claim 5, wherein the act of initiating communication with the user includes the act of sending a message to the user through a wireless communication procedure.

8. The method of claim 5, wherein the act of initiating communication with the user includes the act of sending an email message to the user.

9. The method of claim 5, further comprising the act of: activating a second notification server if the user has not been successfully informed of the scheduled event, wherein the second notification server initiates communication with the user to inform the user of the scheduled event.

10. A method for communicating scheduling information about a future event to a user that operates scheduling software at a client computer on a network, wherein scheduling information for the user is stored at the client computer, the method comprising the acts of:

entering scheduling information at the client computer including the manner in which the user is to be notified of the future event and the time at which the user is to be notified of the future event;

initiating a request over the network to retrieve scheduling information for the user;

accessing the user's scheduling information through programming interface calls to the scheduling software on the client computer;

retrieving at least some of the user's scheduling information;

analyzing the user's scheduling information;

queuing to a storage area at least some of the user's scheduling information that has been analyzed;

accessing the storage area to retrieve the queued scheduling information; and

initiating communication with the user to inform the user of the future event in the manner configured and at the time configured when the user is not at the client computer.

11. The method of claim 10, wherein the act of accessing the user's scheduling information through programming interface calls includes the act of accessing the user's scheduling information through calls to a COM interface.

12. A method for communicating scheduling information to a user that operates scheduling software at a client computer on a network, wherein scheduling information for the user is stored at the client computer, the method comprising the acts of:

initiating a request over the network to retrieve scheduling information for the user;

accessing the user's scheduling information through programming interface calls to the scheduling software on the client computer;

retrieving at least some of the user's scheduling information;

10

analyzing the user's scheduling information;

queuing to a storage area information relating to events in the user's scheduling information;

accessing the storage area to retrieve the queued events; and

initiating communication with the user at or about the time that the queued event is to occur so as to inform the user of a scheduled event when the user is not at the client computer,

wherein the information about how the user is to be notified when an event approaches applies to a subset of all events in the user's scheduling information.

13. The method of claim 12, wherein the act of queuing to a storage area includes the act of queuing to a storage area events that are scheduled for a time within one hour of the act of accessing the user's scheduling information.

14. The method of claim 13, wherein the act of initiating communication with the user includes the act of placing a wireless telephone call to the user.

15. The method of claim 13, wherein the act of initiating communication with the user includes the act of sending a message to the user through a wireless communication procedure.

16. The method of claim 15, wherein the act of initiating communication with the user includes the act of sending a message to a wireless pager.

17. The method of claim 13, wherein the act of initiating communication with the user includes the act of sending an email message to the user.

18. The method of claim 13, further comprising the act of: activating a second notification server if the user has not been successfully informed of the scheduled event, wherein the second notification server initiates communication with the user to inform the user of the scheduled event.

19. In a computer that is connected to at least one notification server corresponding to a means of communication, wherein the computer has access to scheduling information about a future event for a user, the method comprising the acts of:

entering scheduling information including the manner in which the user is to be notified of the future event and the time at which the user is to be notified of the future event;

retrieving the user's scheduling information;

analyzing the user's scheduling information;

queuing the scheduling information if it corresponds to a specified time frame;

accessing the queue to retrieve at least some of the queued scheduling information; and

activating the notification server at or about the time at which the user is to be notified of the future event to thereby initiate communication with the user and inform the user of the future event.

20. The method of claim 19, wherein the act of retrieving the user's scheduling information includes the act of retrieving scheduling information from a remote location.

21. The method of claim 19, wherein the act of retrieving the user's scheduling information includes the act of retrieving scheduling information from a local resource.

22. In a computer that is connected to at least one notification server that is capable of initiating communication with a user, wherein the computer has access to scheduling information for the user, the method comprising the acts of:

11

retrieving the user's scheduling information;
 analyzing the user's scheduling information;
 queuing events in the scheduling information that are
 within a specified time frame by storing them on a
 queue;
 accessing the queue to retrieve one or more queued
 events; and
 activating the notification server at or about the time that
 the queued event is to occur to thereby initiate com-
 munication with the user and inform the user of a
 scheduled event,
 wherein the act of retrieving the user's scheduling infor-
 mation includes the act of retrieving information about
 how the user is to be notified when an event
 approaches, and
 wherein the information about how the user is to be
 notified when an event approaches applies to all events
 in the user's scheduling information.

12

23. The method of claim **22**, wherein the act of initiating communication with the user includes the act of placing a wireless telephone call to the user.

24. The method of claim **22**, wherein the act of initiating communication with the user includes the act of sending a message to the user through a wireless communication procedure.

25. The method of claim **22**, wherein the act of initiating communication with the user includes the act of sending a message to the user through a pager.

26. The method of claim **22**, wherein the act of initiating communication with the user includes the act of sending an email message to the user.

27. The method of claim **22**, further comprising the act of: activating a second notification server if the user has not been successfully informed of the scheduled event, wherein the second notification server initiates communication with the user to inform the user of the scheduled event.

* * * * *

09/218,916

L Number	Hits	Search Text	DB	Time stamp
1	6013	(wireless OR portable) SAME transceiver	USPAT	2003/05/01 15:19
2	1888	event WITH (notification OR notifier)	USPAT	2003/05/01 15:20
3	3	((wireless OR portable) SAME transceiver) SAME (event WITH (notification OR notifier))	USPAT	2003/05/01 15:21
4	30140	event WITH detect\$4	USPAT	2003/05/01 15:22
6	34	((wireless OR portable) SAME transceiver) SAME ((event WITH (notification OR notifier)) OR (event WITH detect\$4))	USPAT	2003/05/01 15:22
5	31436	(event WITH (notification OR notifier)) OR (event WITH detect\$4)	USPAT	2003/05/01 15:24
-	1	5805886.pn.	USPAT	2001/12/31 10:17
-	1	5822583.pn.	USPAT	2001/12/31 10:17
-	2	5805886.pn. OR 5822583.pn.	USPAT	2001/12/31 15:12
-	58522	display ADJ2 device	USPAT	2001/12/31 15:12
-	6820	pager OR beeper	USPAT	2001/12/31 15:12
-	772	(display ADJ2 device) AND (pager OR beeper)	USPAT	2001/12/31 15:13
-	91	709/318.ccls.	USPAT	2001/12/31 15:13
-	0	((display ADJ2 device) AND (pager OR beeper)) AND 709/318.ccls.	USPAT	2001/12/31 15:13
-	284	event ADJ3 notifi\$4	USPAT	2001/12/31 15:13
-	757	event ADJ3 notifi\$6	USPAT	2002/01/07 15:02
-	8	((display ADJ2 device) AND (pager OR beeper)) AND (event ADJ3 notifi\$6)	USPAT	2002/01/02 11:23
-	16019	(LED OR LCD) ADJ2 display	USPAT	2001/12/31 15:20
-	0	(event ADJ3 notifi\$6) SAME ((LED OR LCD) ADJ2 display)	USPAT	2001/12/31 15:21
-	43	(event ADJ3 notifi\$6) AND ((LED OR LCD) ADJ2 display)	USPAT	2001/12/31 15:21
-	5546	((speech ADJ synthesizer) OR audible OR voice) ADJ4 message	USPAT	2002/01/02 11:22
-	109	(display ADJ2 device) AND (event ADJ3 notifi\$6)	USPAT	2002/01/02 11:32
-	4	((speech ADJ synthesizer) OR audible OR voice) ADJ4 message) AND ((display ADJ2 device) AND (event ADJ3 notifi\$6))	USPAT	2002/01/02 11:31
-	1	5822583.pn.	USPAT	2002/01/02 11:32
-	0	5822583.pn. AND (event ADJ3 notifi\$6)	USPAT	2002/01/02 11:32
-	7759	beeper OR pager OR (portable ADJ (transmitter OR transceiver))	USPAT	2002/01/07 15:01
-	45	(event ADJ3 notifi\$6) AND (beeper OR pager OR (portable ADJ (transmitter OR transceiver)))	USPAT	2002/01/07 15:03
-	10	(event ADJ3 notifi\$6) SAME (beeper OR pager OR (portable ADJ (transmitter OR transceiver)))	USPAT	2002/01/07 15:03